This is a postprint version of the following published document:

Capitani, M., Giannone, F., Fichera, S., Sgambelluri, A., Kondepu, K., Kraja, E., Martini, B., Landi, G. y Valcarenghi, L. (2018). Experimental Demonstration of a 5G Network Slice Deployment Through the 5G-Transformer Architecture. In: *2018 European Conference on Optical Communication (ECOC)*.

DOI: https://doi.org/10.1109/ECOC.2018.8535163

Experimental Demonstration of a 5G Network Slice Deployment through the 5G-Transformer Architecture

M. Capitani(1), F. Giannone(2), S. Fichera(2), A. Sgambelluri(2), K. Kondepu(2), E. Kraja(1), B. Martini(2)(3), G. Landi(1), L. Valcarenghi(2)
(1) Nextworks, Pisa, Italy, m.capitani@nextworks.it
(2) Scuola Superiore Sant'Anna, Pisa, Italy; (3) CNIT, Pisa, Italy

Abstract *This demo shows for the first time the capability of the proposed 5G-TRANSFORMER architecture to deploy a 5G network slice in minutes. The slice consists of a mix of physical and virtual mobile network functions deployed in an Openstack environment.*

Introduction

Three main factors are currently shaping future mobile networks. First of all, the concept of slicing has been recently introduced as a method for allowing several virtual networks to share a single physical network[1]. Furthermore, the envisioned 5G network architecture will be heavily based on virtual network function (VNF) representing "a transition from today's network of entities to a network of (virtual) functions"[2]. Finally, the design, the control, and the management of networks will be driven by the latency and capacity required by emerging 5G services (e.g., eMBB---enhanced Mobile Broadband; mMTC---massive Machine Type Communications; URLLC---Ultra-Reliable and Low Latency Communications)[3] and architectures (e.g., the New RAN)[4].

In this scenario, the H2020 5G-TRANSFORMER project (5GT) aims at transforming today's rigid mobile transport networks into a flexible SDN/NFV-based mobile transport and computing platform supporting different verticals (e.g., automotive, e-health, e-industry).

This demo will show how a framework based on the 5GT is capable of deploying a mobile network slice in few minutes. The slice consists of a mix of physical network functions (PNFs) and of VNFs. The latter ones are deployed in Openstack.

State of the Art

Several Standard Developing Organizations (SDOs), industry alliances, and research projects are addressing the definition of a network slice. For example, 3GPP in TR 28.801[5] defines, a network slice instance (NSI) as "a set of network functions and the resources for these network functions which are arranged and configured, forming a complete logical network to meet certain network characteristics". ETSI NFV EVE012[6] establishes the correspondence between a network slice (3GPP) and a network service (ETSI NFV). There, ETSI describes that a Network Function Virtualisation (NFV) Network Service (NFV-NS) can be regarded as a resource-centric view of a network slice, for the cases where a NSI would contain at least one VNF.

NFV[7] ease network slicing by providing slices with multiple instances of the same network function. The exploitation of NFV is foreseen in the New Radio (NR) access technology[4] as well. In NR both the next generation eNB (gNB) and the Next Generation Core (NGC) functions can be virtualised.

To manage and orchestrate virtualised network functions, ETSI is defining a framework (i.e., ETSI Management and Orchestration (MANO)) for the provisioning, management, and orchestration of virtual network services and resources[8].

The demo shows for the first time the capability of the 5GT architecture to deploy a 5G mobile network slice in minutes.

5GT Architecture and Developed Components for the Demo

5GT architecture is based on the ETSI MANO architecture[9]. It includes the 5GT vertical slicer (5GT-VS), the 5GT service orchestrator (5GT-SO), and the 5GT Mobile Transport and Computing Platform (5GT-MTP).

The 5GT-VS is the entry point for the vertical requesting a service and it handles the association of these services with slices as well as network slice management. The 5GT-SO is responsible for end-to-end orchestration of services across multiple domains and exposing them to the 5GT-VS in a unified way. The 5GT-MTP provides, manages, and abstracts the virtual and physical IT and network resources on which service components are deployed.

The demo objective is to experimentally demonstrate for the first time the deployment of a 5G mobile network slice through the 5GT architecture. Thus, this demo is based on an implementation of the 5GT-VS and of the 5GT-SO. The role of the 5GT-MTP in this demo is played by OpenStack, because the additional

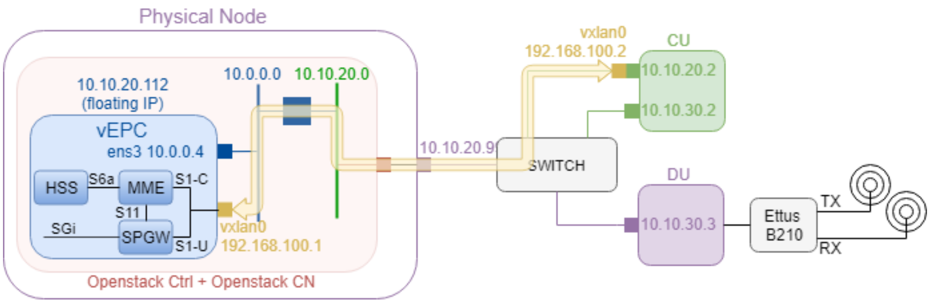

Fig. 1 Demo Setup

functionalities that would be provided by a full implementation of the 5GT-MTP are not needed. The 5GT-VS prototype is a Java application based on the Spring framework, exposing its functionality through a RESTful HTTP API and a web GUI. The application is disaggregated, with its several components, i.e. slice lifecycle manager, translator, arbitrator and SouthBound (SB) drivers, communicating through Simple Message Queue Protocol (SMQP).

The 5GT-SO implementations provides the NFVO and VNFM functionality as described in[6], extended with functionality for managing PNF-containing Network Services. To do that, an additional component called Physical Network Function Manager (PNFM) has been implemented, which takes care of configuring the PNFs with the necessary parameters through RESTful HTTP messages.

Demo Setup and Workflow

The demo setup is described in Fig. 1. The open source OAI platform[10] is utilised as mobile network software. OAI provides an implementation of few New RAN functional splits (as defined in 3GPP TR 38.801[4]), where, the evolved NodeB (eNB) functions are decoupled into two new network entities such as Central Unit (CU), where the base-band processing is centralized, and Distributed Unit (DU), where the RF processing is left at the antenna. In the demonstration, as shown in Fig. 1, both DU and CU are deployed as PNF and they utilise Option 7-1 (i.e., intra-PHY) functional split. The OAI core is utilised for implementing the EPC functions. OAI EPC contains the implementation of the following network elements: the Serving Gateway (S-GW), the PDN Gateway (PDN GW), the Mobile Management Entity (MME) and the Home Subscriber Server (HSS). All these OAI core elements can be deployed as individual VNF elements in a virtualised environment or can also be deployed as bundle vEPC VNF. In the demonstration, the bundle vEPC VNF is utilised. The bundle vEPC VNF is deployed in an OpenStack environment (Ocata). OpenStack is deployed as a single node that includes both the controller (Ctrl) and the compute node (CN). In Openstack two networks are defined: the Openstack private network with address 10.0.0.0/24 and the Openstack public network with address 10.10.20.0/24. The vEPC VNF ens3 interface is assigned an IP address (10.0.0.4) of the Openstack private network. A floating IP (10.10.20.112) is, then, generated from the pool of the Openstack public network addresses and it is mapped to the vEPC VNF ens3 interface address. The floating IP address allows vEPC

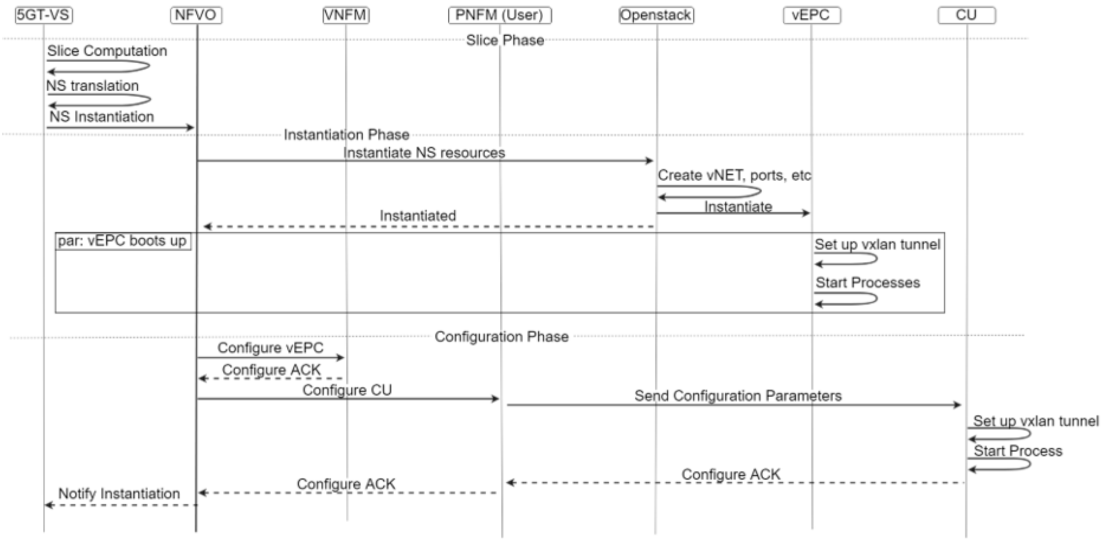

Fig. 2 Demo Workflow

VNF reachability. As shown in Fig. 1, the vEPC VNF is communicating the CU PNF, the CU PNF communicates with the DU PNF, and the User Equipment (UE) is connected to the DU PNF, by means of Universal Software Radio Peripherals (USRPs) Ettus B210.
If the vEPC VNF and CU PNF are in different IP sub networks, a Virtual eXtensible LAN (VXLAN)[11] shall be configured for the data plane interconnection. In this demonstration, because of Openstack configurations, the floating IP is not listed in the vEPC VNF IP addresses. Thus, it cannot be used in the OAI core configuration files of the vEPC VNF. Therefore, even if the vEPC VNF floating IP and the CU PNF IP (10.10.20.2) are in the same IP sub networks, the VXLAN tunnel is established between such network entities. In this way, the VXLAN interface (vxlan0) IP address (192.168.100.1) in the vEPC VNF is used in the related OAI core configuration files and for connecting it to the CU PNF, where a VXLAN interface (vxlan0) IP address (192.168.100.2) is set. At the vEPC VNF side, the configuration of VXLAN with the fixed remote IP of CU PNF is automated by startup scripts. At the CU PNF side, during the instantiation phase of NFVO life cycle event, the NFVO provides the floating IP of vEPC VNF to create the VXLAN.

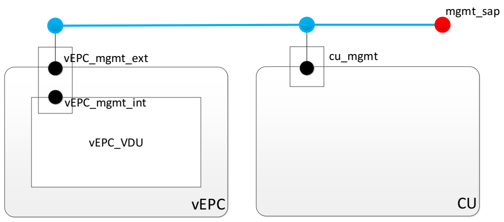

Fig. 3: Representation of the Network Service

The demo workflow is described in Fig. 2. The demo is started by requesting a mobile service at the 5GT-VS. The component translates the service request into a mobile-capable slice, and instantiates a Network Service (see Fig. 3) implementing such a slice through the 5GT-SO.
The 5GT-SO then starts the Instantiation process: at first it requests the instantiation of a vEPC VM to OpenStack (acting as 5GT-MTP). While booting, the vEPC VM creates one end of the VXLAN tunnel and starts the vEPC component processes (MME, HSS, S/PGW). After the instantiation of the VM is notified back to the NFVO, it starts the configuration phase. First it configures the vEPC (in this particular demo, no configuration needs to be applied) then it requests to the PNFM to configure the CU (which is represented in the Network Service as a PNF). The PNFM sends a message to the CU containing the IP of the vEPC, so that the CU can instantiate the other half of the VXLAN tunnel and establish the communication with the vEPC.

The tentative deployment of the demo at the conference premises is the following: UEs, one USPR, DU (one mini-PC), CU (one mini-PC), SWITCH will be at the premises while the vEPC, Openstack, 5GT-VS, 5GT-SO will be deployed in the ARNO testbed in Pisa. Thus, visitors can access Internet through the local antenna while the SGi interface will be at the ARNO tesbed premises.

Performance Parameters

The considered performance parameter will be the slice/service delivery time (SDT). The SDT is the time elapsing between the mobile network slice request and the successful slice delivery. Such time will be measure by running a ping command from the UE to a website. The ping is started contemporarily to the slice request. The SDT corresponds to the time when the first ping receives successful reply.

Conclusions

This paper demonstrated for the first time the deployment of a 5G network slice through the 5G-Transformer Architecture. In minutes visitors are capable of connecting to the mobile slice and connect to the Internet.

Acknowledgements

This work was supported by the EC through the Horizon 2020 5G-TRANSFORMER project (grant agreement 761536).

References

[1] NGMN Alliance, “Description of Network Slicing Concept”, NGMN 5G P1 Requirements & Architecture Work Stream End-to-End Architecture, v1, Jan. 13, 2016
[2] 5G PPP, "View on 5g architecture," White Paper, https://5gppp.eu/white-papers (accessed 10/07/2017).
[3] Recommendation ITU-R M.2083-0, “IMT Vision – Framework and overall objectives of the future development of IMT for 2020 and beyond”, 2015
[4] 3GPP TR 38.801, “Study on new radio access technology: Radio access architecture and interfaces (Release 14)”; V14.0.0 (2017-03).
[5] “3GPP; Technical Specification Group Services and System Aspects; Telecommunication management; Study on management and orchestration of network slicing for next generation network (Release 15)”, 3GPP TR 28.801 V15.1.0 (2018-01)
[6] ETSI GR NFV-EVE 012, “Network Functions Virtualisation (NFV) Release 3; Evolution and Ecosystem; Report on Network Slicing Support with ETSI NFV Architecture Framework” v3.1.1, December 2017.
[7] ETSI GS NFV 002, “Network Functions Virtualisation (NFV); Architectural Framework”, V1.1.1 (2013-10)
[8] ETSI GS NFV-MAN 001 V1.1.1 (2014-12), “Network Functions Virtualisation (NFV); Management and Orchestration”.
[9] 5G-TRANSFORMER D2.1, “Definition of the Mobile Transport and Computing Platform”, http://5g-transformer.eu/
[10] http://www.openairinterface.org
[11] M. Mahalingam *et al.*, "Virtual extensible local area network (VXLAN): A framework for overlaying virtualized layer 2 networks over layer 3 networks", IETF. RFC 7348. August, 2014.